REMARKS

Applicant respectfully requests the Examiner's reconsideration of the present application.

Claims 1-9, 11-15, 17-23, 25-27, 29-35, 37-50 are pending in the present application.

Claims 1-9, 11-15, 17-23, 25-27, 29-33, 35, 38, 39, 41, and 44-50 are rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6,707,844 ("Imaizumi") and U.S. Patent No. 6,212,222 ("Okubo").

Claims 34, 37, 40, 42, and 43 are rejected under 35 U.S.C. §103(a) as being unpatentable over Imaizumi in view of Okubo and U.S. Patent No. 6,490,316 ("Motegi").

Claims 14, 26, 33, 39, and 46 have been amended.

Support for amended claims 14, 26, 33, 39, and 46 can be found at pages 19-28, and Figures 6-10 in the Drawings.

It is submitted that Imaizumi, Okubo, and Motegi do not render claims 1-9, 11-15, 17-23, 25-27, 29-35, 37-50 unpatentable under 35 U.S.C. §103(a).

Imaizumi includes a disclosure of a synchronous circuit and a receiver in which a long code for use in communication can be determined in a short time, and circuit scale can be reduced. Upon receiving input of a data stop signal from a control unit, a matched filter continues to hold the signal held at the time, performs a product sum operation of a spreading code successively inputted from a spreading code generator and the held signal, and successively outputs correlation signals in the synchronous circuit and the receiver (See Imaizumi Abstract).

Okubo includes a disclosure of a delay correction unit and a multiplexer that calculates the multiplexed square correlation value by matching the peak values and adding the square correlation values of all the multiplexed signals. The multiplexed signals are the signals obtained by multiplexing an RF signal in such manner that different specific delay times are respectively given to parallel transmission information sequences multiplied by identical spread codes.

Therefore, code synchronization can be realized by using only one correlator for both the orthogonal and an in-phase components (See Okubo Abstract).

Motegi includes a disclosure of an apparatus for obtaining correlation by despreading a spread symbol with a spreading code is provided with a symbol storage section for storing symbol and a plurality of correlation calculators each of which executes a correlation calculation to obtain correlation by despreading the symbol data with a spreading code. The apparatus switches a spreading code to be provided to each of the plurality of correlation calculators individually so that each of the plurality of correlation calculators executes correlation calculation with a respective different spreading code and holds a same spreading code until correlation calculations of a plurality of symbols are finished (See Motegi Abstract).

It is submitted that Imaizumi, Okubo, and Motegi do not teach or suggest determining first intermediate correlation values for a first plurality of sample sequences during a first clock cycle, determining second intermediate correlation values for the first plurality of sample sequences during a second clock cycle, determining correlation outputs for the first plurality of sample sequences from the first and second intermediate correlation values, and determining a synchronization point that identifies an amount of delay incurred from transmission of the sample sequences from the correlation outputs.

The Office Action mailed 8/7/2007 states in part that

> Regarding claim 1, Imaizumi teaches a method for managing a code sequence (see Fig. 8), comprising: determining first intermediate correlation values (first two outputs of the elements 42) for a first plurality of sample sequences (from DATA received) during a first clock cycle (see spreading code 1 during the first clock cycle in Fig. 4) determining second intermediate correlation values (second two outputs of the elements 42) for the first plurality of sample sequences (from DATA received) during a second clock cycle (See spreading code 2 during second clock cycle in Fig. 4); determining correlation outputs (output of element 43 in Fig. 8) for the first plurality of sample sequences (for the DATA received) from the first and second intermediate correlation values (input to element 43 is based on the elements 42). And although Imaizumi teaches the correlator performing the steps above, does not explicitly further teach determining a

> synchronization point that identifies an amount of delay incurred from transmission of the sample sequences from the correlation outputs. Okubo also teaches, in the same field of endeavor, a correlating step (see Fig. 1) further coupled to a code synchronization point detector (element 132) of determining a synchronization point (output of element 132) that identifies an amount of delay incurred from transmission of the sample sequences from the correlation outputs (see Fig. 7 of notation and explanation of **).

(8/7/2007 Office Action, pp. 2-3).

On the contrary, Imaizumi discloses a matched filter 4'' that includes a plurality of hold circuits 41. A plurality of multipliers 42 receive data from the hold circuits 41 and generate products from the data in the hold circuits 41 and spreading code in a register 47. The multipliers 42 are directly connected to an adder 43. The products from multipliers 42 (what the Office refers to as the "intermediate correlation values") are transmitted directly to adder 43 (see Imaizumi column 17, lines 32-49, and Figure 8). Applicant submits that since the adder 43 is directly connected to multipliers 42, the adder 43 is capable of only summing products from the multipliers 42 that are generated at the same clock cycle. Thus, Imaizumi does not teach or suggest determining correlation outputs for a first plurality of sample sequences from first intermediate correlation values determined from a first clock cycle and second intermediate correlation values determined from a second clock cycle.

Furthermore, Okubo discloses a code synchronization point detector 132 that estimates a code synchronization point of the entire system based on a time when a multiplexed square correlation value exhibits a maximum value (see Okubo column 7, lines 52-59). The code synchronization point is described as being "obtained when no delay times are not given by the spread modulator 22(1) to 22(n) in the transmission device" (see Okumbo column 5, lines 27-30). Thus, Okubo does not teach or suggest determining a synchronization point where the synchronization point identifies an amount of delay incurred from transmission of sample sequences.

Motegi only discloses a correlation detection apparatus and CDMA receiving apparatus. Motegi does not teach or suggest determining first intermediate correlation values for a first plurality of sample sequences during a first clock cycle, determining second intermediate correlation values for the first plurality of sample sequences during a second clock cycle, determining correlation outputs for the first plurality of sample sequences from the first and second intermediate correlation values, and determining a synchronization point that identifies an amount of delay incurred from transmission of the sample sequences from the correlation outputs.

In contrast, claim 1 states

> A method for managing a code sequence, comprising:
> determining first intermediate correlation values for a first plurality of sample sequences during a first clock cycle;
> determining second intermediate correlation values for the first plurality of sample sequences during a second clock cycle;
> determining correlation outputs for the first plurality of sample sequences from the first and second intermediate correlation values; and
> determining a synchronization point that identifies an amount of delay incurred from transmission of the sample sequences from the correlation outputs

(Claim 1) (Emphasis added).

Claim 6 includes similar limitations. Given that claims 2-5 and 47 depend from claim 1, and claims 7-9, 11-13, and 48-49 depend from claim 6, it is likewise submitted that claims 2-5, 7-9, 11-13, and 47-49 are also patentable under 35 U.S.C. §103(a) over Imaizumi, Okubo, and Motegi.

Applicant further submits that Imaizumi, Okubo, and Motegi do not teach or suggest organizing a code sequence, having L contiguous coefficients, into L/n contiguous code sequence groups having n coefficients each, where n is greater than 1, selecting a number of sample sequences to process in parallel where each of the sample sequences has contiguous sample values from a received sample, organizing contiguous sample values from each of a first set of contiguous sample sequences to process in parallel into a first set of contiguous sample sequence

groups, processing coefficients in each of the code sequence groups in parallel with corresponding sample values in corresponding sample sequence groups from the first set of sample sequences, where each of the code sequence groups is processed during a different clock cycle, determining a correlation output for each of the sample sequences, and determining a synchronization point that identifies an amount of delay incurred from transmission of the sample sequences from the correlation output.

The Office Action mailed 8/7/2007 states in part that

> Regarding claim 14, the claim is rejected as applied to claim 1. Imaizumi further teaches wherein the code sequence having L/n groups (n=1 and L is the number of Spreading code in element 47 in Fig. 8).

(8/7/2007 Office Action, p. 6).

Applicant disagrees with the Office. The Office appears to be equating the spreading code disclosed in Imaizumi with the code sequence recited in applicant's claims. If Imaizumi discloses a "code sequence having L/n groups ... [where] n=1 and L is the number of Spreading code in element 47", then the register 47 would be capable of storing all of the coefficients of all the code sequence groups at the same time (see Imaizumi Figure 8). When the register 47 stores all of the coefficients from all of the code sequence group at the same time, more than one of the L/n code sequence groups would be processed at the same clock cycle (see Imaizumi Figure 8 47, 42, 41). Thus, Imaizumi does not teach or suggest processing coefficients in each of the code sequence groups in parallel with corresponding sample values in corresponding sample sequence groups where each of the code sequence groups is processed during a different clock cycle.

Furthermore, as stated above, Okubo discloses a code synchronization point detector 132 that estimates a code synchronization point of the entire system based on a time when a multiplexed square correlation value exhibits a maximum value (see Okubo column 7, lines 52-59). The code synchronization point is described as being "obtained when no delay times are not given by the spread modulator 22(1) to 22(n) in the transmission device" (see Okumbo

column 5, lines 27-30). Thus, Okubo does not teach or suggest determining a synchronization point where the synchronization point identifies an amount of delay incurred from transmission of sample sequences.

Motegi only discloses a correlation detection apparatus and CDMA receiving apparatus. Motegi does not teach or suggest organizing a code sequence, having L contiguous coefficients, into L/n contiguous code sequence groups having n coefficients each, where n is greater than 1, selecting a number of sample sequences to process in parallel where each of the sample sequences has contiguous sample values from a received sample, organizing contiguous sample values from each of a first set of contiguous sample sequences to process in parallel into a first set of contiguous sample sequence groups, processing coefficients in each of the code sequence groups in parallel with corresponding sample values in corresponding sample sequence groups from the first set of sample sequences, where each of the code sequence groups is processed during a different clock cycle, determining a correlation output for each of the sample sequences, and determining a synchronization point that identifies an amount of delay incurred from transmission of the sample sequences from the correlation output.

In contrast, claim 14 states

> A method for managing a code sequence, comprising:
> organizing the code sequence, having L contiguous coefficients, into L/n contiguous code sequence groups having n coefficients each, where n is greater than 1;
> selecting a number of sample sequences to process in parallel where each of the sample sequences has contiguous sample values from a received sample;
> organizing contiguous sample values from each of a first set of contiguous sample sequences to process in parallel into a first set of contiguous sample sequence groups;
> processing coefficients in each of the code sequence groups in parallel with corresponding sample values in corresponding sample sequence groups from the first set of sample sequences, where each of code sequence groups is processed during a different clock cycle;
> determining a correlation output for each of the sample sequences; and

> determining a synchronization point that identifies an amount of delay incurred from transmission of the sample sequences from the correlation output.

(Claim 14) (Emphasis added).

Claims 26, 33, 39, and 46 include similar limitations.

Given that claims 15, 17-23, 25, and 50 depend directly or indirectly from claim 14, claims 27, and 29-32 depend directly or indirectly from claim 26, claims 34-35, and 37-38 depend directly or indirectly from claim 33, and claims 40-45 depend directly and indirectly from claim 39, it is likewise submitted that claims 15, 17-23, 25, 27, 29-32, 34-35, 37-38, 40-45, and 50 are also patentable under 35 U.S.C. §103(a) over Imaizumi, Okubo, and Motegi.

Applicant further submits that Imaizumi, Okubo, and Motegi do not teach or suggest organizing a code sequence, having L contiguous coefficients, into L/n contiguous code sequence groups having n coefficients each, selecting a number of sample sequences to process in parallel where each of the sample sequences has contiguous sample values from a received sample, organizing contiguous sample values from each of a first set of contiguous sample sequences to process in parallel into a first set of contiguous sample sequence groups, processing coefficients in each of the code sequence groups in parallel with corresponding sample values in corresponding sample sequence groups from the first set of sample sequences, where each of the code sequence groups is processed during a different clock cycle, and processing coefficients comprises processing coefficients for L/n clocks.

The Office Action mailed 8/7/2007 states in part that

> Regarding claim 25, Imaizumi further teaches wherein processing coefficients comprises processing coefficients for L/n clocks (n=1 and L is the number of Spreading code in element 47 in Fig. 8, hence processes for L/n clocks).

(8/7/2007 Office Action, p. 8).

Applicant disagrees with the Office. As stated above, the Office appears to be equating the spreading code disclosed in Imaizumi with the code sequence recited in applicant's claims. If

Imaizumi discloses a "code sequence having L/n groups ... [where] n=1 and L is the number of Spreading code in element 47", then the register 47 would be capable of storing all of the coefficients of all the code sequence groups at the same time (see Imaizumi Figure 8). When the register 47 stores all of the coefficients from all of the code sequence group at the same time, more than one of the L/n code sequence groups would be processed at the same clock cycle (see Imaizumi Figure 8 47, 42, 41). Applicant submits that Imaizumi appears to provide no disclosure of processing coefficients in each of the code sequence groups for L/n clocks. Applicant requests that the Office clarify how it believes that the matched filter 4'' processes coefficients in each code sequence groups for L/n clocks especially when each of the L/n code sequence groups has only 1 (n=1) coefficient.

Okubo only discloses an initial acquisition circuit. Okubo does not teach or suggest organizing a code sequence, having L contiguous coefficients, into L/n contiguous code sequence groups having n coefficients each, selecting a number of sample sequences to process in parallel where each of the sample sequences has contiguous sample values from a received sample, organizing contiguous sample values from each of a first set of contiguous sample sequences to process in parallel into a first set of contiguous sample sequence groups, processing coefficients in each of the code sequence groups in parallel with corresponding sample values in corresponding sample sequence groups from the first set of sample sequences, where each of the code sequence groups is processed during a different clock cycle, and processing coefficients comprises processing coefficients for L/n clocks.

Motegi only discloses a correlation detection apparatus and CDMA receiving apparatus. Motegi does not teach or suggest organizing a code sequence, having L contiguous coefficients, into L/n contiguous code sequence groups having n coefficients each, selecting a number of sample sequences to process in parallel where each of the sample sequences has contiguous sample values from a received sample, organizing contiguous sample values from each of a first set of contiguous sample sequences to process in parallel into a first set of contiguous sample

sequence groups, processing coefficients in each of the code sequence groups in parallel with corresponding sample values in corresponding sample sequence groups from the first set of sample sequences, where each of the code sequence groups is processed during a different clock cycle, and processing coefficients comprises processing coefficients for L/n clocks.

In contrast, claim 25 states

> The method of Claim 14, wherein processing coefficients comprises processing coefficients for L/n clocks.

(Claim 25) (Emphasis added).

Claims 32 and 45 include similar limitations.

Applicant further submits that Imaizumi, Okubo, and Motegi do not teach or suggest a correlator unit that includes a plurality of code sequence registers that store coefficients from a code sequence group having n coefficients, the plurality of code sequence registers storing coefficients from one code sequence group of L/n code sequence groups at a time, where L is the number of coefficients in a code sequence, where n is greater than 1, a plurality of sample registers that store sample values from a plurality of sample sequences that are processed in parallel, a processing unit that processes coefficients in each of the plurality of code sequence groups in the plurality of code sequence registers in parallel with corresponding sample values in corresponding sample sequence groups from a first plurality of sample sequences in the plurality of sample registers, where each of the code sequence groups is processed to generate intermediate correlation values during a different clock cycle, an accumulation unit that generates a correlation output for each of the sample sequences from the intermediate correlation values generated during the different clock cycles.

On the contrary Imaizumi discloses an adder 43. The adder 43 sums up values generated from multipliers 42 during a same clock cycle. The values generated from the multipliers 42 are not intermediate correlation values generated during different clock cycles. Thus, applicant submits that adder 43 is not an accumulation unit that determines

a correlation output from each of the sample sequences from the intermediate correlation values generated during different clock cycles.

Okubo only discloses an initial acquisition circuit. Okubo does not teach or suggest a correlator unit that includes a plurality of code sequence registers that store coefficients from a code sequence group having n coefficients, the plurality of code sequence registers storing coefficients from one code sequence group of L/n code sequence groups at a time, where L is the number of coefficients in a code sequence, where n is greater than 1, a plurality of sample registers that store sample values from a plurality of sample sequences that are processed in parallel, a processing unit that processes coefficients in each of the plurality of code sequence groups in the plurality of code sequence registers in parallel with corresponding sample values in corresponding sample sequence groups from a first plurality of sample sequences in the plurality of sample registers, where each of the code sequence groups is processed to generate intermediate correlation values during a different clock cycle, an accumulation unit that generates a correlation output for each of the sample sequences from the intermediate correlation values generated during the different clock cycles.

Motegi only discloses a correlation detection apparatus and CDMA receiving apparatus. Motegi does not teach or suggest a correlator unit that includes a plurality of code sequence registers that store coefficients from a code sequence group having n coefficients, the plurality of code sequence registers storing coefficients from one code sequence group of L/n code sequence groups at a time, where L is the number of coefficients in a code sequence, where n is greater than 1, a plurality of sample registers that store sample values from a plurality of sample sequences that are processed in parallel, a processing unit that processes coefficients in each of the plurality of code sequence groups in the plurality of code sequence registers in parallel with corresponding sample values in corresponding sample sequence groups from a first plurality of sample sequences in the plurality of sample registers, where each of the code sequence groups is processed to generate intermediate correlation values during a different clock cycle, an

accumulation unit that generates a correlation output for each of the sample sequences from the intermediate correlation values generated during the different clock cycles.

In contrast, claim 33 states

> A correlator unit, comprising:
> a plurality of code sequence registers that store coefficients from a code sequence group having n coefficients, the plurality of code sequence registers storing coefficients from one code sequence group of L/n code sequence groups at a time, where L is the number of coefficients in a code sequence, where n is greater than 1;
> a plurality of sample registers that store sample values from a plurality of sample sequences that are processed in parallel;
> a processing unit that processes coefficients in each of the plurality of code sequence groups in the plurality of code sequence registers in parallel with corresponding sample values in corresponding sample sequence groups from a first plurality of sample sequences in the plurality of sample registers, where each of the code sequence groups is processed to generate intermediate correlation values during a different clock cycle;
> an accumulation unit that generates a correlation output for each of the sample sequences from the intermediate correlation values generated during the different clock cycles; and
> a correlation output processor that determines a synchronization point that identifies an amount of delay incurred from transmission of the sample sequences from the correlation output.

(Claim 33) (Emphasis added).

Claims 39 and 46 include similar limitations. Given that claims 34-35, 37-38 depend from claim 33, and claims 40-45 depend from claim 39, it is submitted that claims 34-35, 37-38, and 40-45 are also patentable under 35 U.S.C. §103(a) over Imaizumi, Okubo, and Motegi.

In view of the amendments and arguments set forth herein, it is respectfully submitted that the applicable rejections and have been overcome. Accordingly, it is respectfully submitted that claims 1-9, 11-15, 17-23, 25-27, 29-35, 37-50 should be found to be in condition for allowance.

The Examiner is invited to telephone Applicant's attorney (217-377-2500) to facilitate prosecution of this application.

If any additional fee is required, please charge Deposit Account No. 50-1624.

Respectfully submitted,

Dated: August 18, 2008

Lawrence M. Cho
Attorney for Applicant
Registration No. 39,942

P.O. Box 2144
Champaign, IL 61825
(217) 377-2500